Results for "((graphical programming)<in>metadata)" e-mail
Your search matched **152** of **1198558** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

((graphical programming)<in>metadata)

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Select Article Information View: **1-25** | 26-5

1. **The applicability of the visual programming language LabVIEW to large r applications**
Jamal, R.; Wenzel, L.;
Visual Languages, Proceedings., 11th IEEE International Symposium on
5-9 Sept. 1995 Page(s):99 - 106
Digital Object Identifier 10.1109/VL.1995.520791
AbstractPlus | Full Text: PDF(996 KB) IEEE CNF

2. **Developing a visual component library for a graphical programming platf orientation**
Wang, O.L.; Huang, J.;
Aerospace and Electronic Systems Magazine, IEEE
Volume 17, Issue 6, June 2002 Page(s):16 - 19
Digital Object Identifier 10.1109/MAES.2002.1010116
AbstractPlus | References | Full Text: PDF(296 KB) IEEE JNL

3. **Developing visual component library for a graphical programming platfo orientation**
Wang, L.L.;
AUTOTESTCON Proceedings, 2001. IEEE Systems Readiness Technology C
20-23 Aug. 2001 Page(s):672 - 678
Digital Object Identifier 10.1109/AUTEST.2001.949451
AbstractPlus | Full Text: PDF(328 KB) IEEE CNF

4. **A graphical programming environment for simulation of control and sign systems**
Waknis, P.; Karsai, G.; Sztipanovits, J.;
Southeastcon '92, Proceedings., IEEE
12-15 April 1992 Page(s):447 - 450 vol.1
Digital Object Identifier 10.1109/SECON.1992.202388
AbstractPlus | Full Text: PDF(340 KB) IEEE CNF

5. **Use of a VXI algorithmic closed loop controller and a graphical programn control a film process tester**
Baldassari-Mather, L.;
Instrumentation and Measurement Technology Conference, 1998. IMTC/98. C
Proceedings. IEEE
Volume 1, 18-21 May 1998 Page(s):73 - 76 vol.1

Digital Object Identifier 10.1109/IMTC.1998.679671

AbstractPlus | Full Text: PDF(308 KB) IEEE CNF

6. **Graphical programming environment for ATLAS**
O'Toole, K.;
AUTOTESTCON '97. 1997 IEEE Autotestcon Proceedings
22-25 Sept. 1997 Page(s):488 - 491
Digital Object Identifier 10.1109/AUTEST.1997.633662

AbstractPlus | Full Text: PDF(320 KB) IEEE CNF

7. **Dynamic simulation of robot manipulators using graphical programming**
Sun, S.-S.; Meng, Q.-H.M.;
Communications, Computers, and Signal Processing, 1995. Proceedings. IEEI Conference on
17-19 May 1995 Page(s):521 - 524
Digital Object Identifier 10.1109/PACRIM.1995.519584

AbstractPlus | Full Text: PDF(208 KB) IEEE CNF

8. **Computer-based automated test measurement system for determining m characteristics of switched reluctance motors**
Cheok, A.D.; Ertugrul, N.;
Instrumentation and Measurement, IEEE Transactions on
Volume 50, Issue 3, June 2001 Page(s):690 - 696
Digital Object Identifier 10.1109/19.930441

AbstractPlus | References | Full Text: PDF(140 KB) IEEE JNL

9. **A graphical programming environment for message passing programs**
Kacsuk, P.; Dozsa, G.; Fadgyas, T.;
Software Engineering for Parallel and Distributed Systems, 1997. Proceedings International Workshop on
17-18 May 1997 Page(s):210 - 219
Digital Object Identifier 10.1109/PDSE.1997.596840

AbstractPlus | Full Text: PDF(1020 KB) IEEE CNF

10. **Diagrammatic-graphical programming languages and DoD-STD-2167A**
Bragg, S.D.; Driskill, C.G.;
AUTOTESTCON '94. IEEE Systems Readiness Technology Conference. 'Cos Support Into the Next Century', Conference Proceedings.
20-22 Sept. 1994 Page(s):211 - 220
Digital Object Identifier 10.1109/AUTEST.1994.381508

AbstractPlus | Full Text: PDF(388 KB) IEEE CNF

11. **Building and using graphical programming tools for the IEC 1131-3 stanc**
Juer, J.; Oliver, J.J.;
Advances in Software Engineering for PLC
14 Oct 1993 Page(s):5/1 - 5/4

AbstractPlus | Full Text: PDF(224 KB) IEE CNF

12. **A novel switched reluctance motor drive with optical graphical programn**
Pillay, P.; Yaguang Liu; Durham, O.G.;
Industrial Electronics, IEEE Transactions on
Volume 47, Issue 4, Aug. 2000 Page(s):915 - 919
Digital Object Identifier 10.1109/41.857972

AbstractPlus | References | Full Text: PDF(180 KB) IEEE JNL

13. **Graphical programming using UML and SDL**
Bjorkander, M.;
Computer

Volume 33, Issue 12, Dec. 2000 Page(s):30 - 35
Digital Object Identifier 10.1109/2.889090

AbstractPlus | References | Full Text: PDF(348 KB) IEEE JNL

14. **Digital signal processing system design: using LabVIEW and TMS320C6(**
Kehtarnavaz, N.; Kim, N.; Panahi, I.;
Digital Signal Processing Workshop, 2004 and the 3rd IEEE Signal Processing Workshop. 2004 IEEE 11th
1-4 Aug. 2004 Page(s):10 - 14
Digital Object Identifier 10.1109/DSPWS.2004.1437901

AbstractPlus | Full Text: PDF(496 KB) IEEE CNF

15. **Framework design and implementation for virtual instrument component**
Xu Xiao-liang; Wang Le-yu; Zhou Hong;
AUTOTESTCON 2003. IEEE Systems Readiness Technology Conference. Pr
22-25 Sept. 2003 Page(s):541 - 545
Digital Object Identifier 10.1109/AUTEST.2003.1243629

AbstractPlus | Full Text: PDF(515 KB) IEEE CNF

16. **System modeling using virtual instruments**
Korrapati, R.; Anderson, J.A.; Swain, N.K.; Swain, M.;
SoutheastCon, 2002. Proceedings IEEE
5-7 April 2002 Page(s):121 - 126
Digital Object Identifier 10.1109/.2002.995570

AbstractPlus | Full Text: PDF(550 KB) IEEE CNF

17. **The role of graphical programming languages in teaching DSP**
Amir, N.;
Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2 International Conference on
Volume 6, 5-9 June 2000 Page(s):3514 - 3517 vol.6
Digital Object Identifier 10.1109/ICASSP.2000.860159

AbstractPlus | Full Text: PDF(476 KB) IEEE CNF

18. **A novel switched reluctance motor drive with optical graphical programn**
Pillay, P.; Liu, Y.; Durham, O.G.;
Industrial Electronics Society, 1999. IECON '99 Proceedings. The 25th Annual the IEEE
Volume 3, 29 Nov.-3 Dec. 1999 Page(s):1121 - 1126 vol.3
Digital Object Identifier 10.1109/IECON.1999.819365

AbstractPlus | Full Text: PDF(484 KB) IEEE CNF

19. **Virtual instrumentation with graphical programming for enhanced detect monitoring of partial discharges**
Wahidabanu, R.S.D.; Panneer Selvam, M.A.; Udaya Kumar, K.;
Electrical Insulation Conference, 1997, and Electrical Manufacturing & Coil Wi Conference. Proceedings
22-25 Sept. 1997 Page(s):291 - 296
Digital Object Identifier 10.1109/EEIC.1997.651103

AbstractPlus | Full Text: PDF(352 KB) IEEE CNF

20. **MPL-a graphical programming environment for matrix processing based constraints**
Yeung, R.;
Visual Languages, 1988., IEEE Workshop on
10-12 Oct. 1988 Page(s):137 - 143
Digital Object Identifier 10.1109/WVL.1988.18021

AbstractPlus | Full Text: PDF(472 KB) IEEE CNF

21. **A LabVIEW approach to instrumentation for the TFTR bumper limiter alig**
Skelly, G.N.; Owens, D.K.;
Fusion Engineering, 1991. Proceedings., 14th IEEE/NPSS Symposium on
30 Sept.-3 Oct. 1991 Page(s):765 - 768 vol.2
Digital Object Identifier 10.1109/FUSION.1991.218735
AbstractPlus | Full Text: PDF(284 KB) **IEEE CNF**

22. **Three-dimensional graphical programming in CAEL**
Van Reeth, F.; Flerackers, E.;
Visual Languages, 1993., Proceedings 1993 IEEE Symposium on
24-27 Aug. 1993 Page(s):389 - 391
Digital Object Identifier 10.1109/VL.1993.269554
AbstractPlus | Full Text: PDF(396 KB) **IEEE CNF**

23. **Innovation in teaching mechanical engineering applications**
Abu Zeid, O.A.; Tanyel, M.M.;
Frontiers in Education Conference, 1994. Twenty-fourth Annual Conference. F
2-6 Nov. 1994 Page(s):82 - 86
Digital Object Identifier 10.1109/FIE.1994.580474
AbstractPlus | Full Text: PDF(900 KB) **IEEE CNF**

24. **An object-oriented optimization system**
Cunningham, G.S.; Hanson, K.M.; Jennings, G.R., Jr.; Wolf, D.R.;
Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conferenc
Volume 3, 13-16 Nov. 1994 Page(s):826 - 830 vol.3
Digital Object Identifier 10.1109/ICIP.1994.413773
AbstractPlus | Full Text: PDF(356 KB) **IEEE CNF**

25. **GRAPE: a CASE tool for digital signal parallel processing**
Lauwereins, R.; Engels, M.; Peperstraete, J.; Steegmans, E.; Van Ginderdeure
ASSP Magazine, IEEE [see also IEEE Signal Processing Magazine]
Volume 7, Issue 2, April 1990 Page(s):32 - 43
Digital Object Identifier 10.1109/53.53031
AbstractPlus | Full Text: PDF(1108 KB) **IEEE JNL**

View: 1-25 | 26-5

Terms used **graphical programming** Found **71,274** of **158,639**

Sort results by relevance
Display results expanded form
Save results to a Binder
Search Tips
Open results in a new window
Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown Relevance scale

1 Cg: a system for programming graphics hardware in a C-like language
William R. Mark, R. Steven Glanville, Kurt.Akeley, Mark J. Kilgard
July 2003 **ACM Transactions on Graphics (TOG)**, Volume 22 Issue 3
Full text available: pdf(2.57 MB) Additional Information: full citation, abstract, references, citings, index terms

The latest real-time graphics architectures include programmable floating-point vertex and fragment processors, with support for data-dependent control flow in the vertex processor. We present a programming language and a supporting system that are designed for programming these stream processors. The language follows the philosophy of C, in that it is a hardware-oriented, general-purpose language, rather than an application-specific shading language. The language includes a variety of facilitie ...

2 A recent radical graphical approach to programming
Alfred L. McKinney
June 2003 **Journal of Computing Sciences in Colleges**, Volume 18 Issue 6
Full text available: pdf(44.13 KB) Additional Information: full citation, abstract, references, index terms

This paper provides a very brief overview of the history of graphical approaches to programming. A more detailed description is given of SoftWIRE, a recent radical graphical approach to programming, which has much potential. SoftWIRE utilizes Visual Basic (VB) in the Integrated Development Environment (IDE) Microsoft Visual Studio. NET. The description of SoftWIRE provides details, advantages, and program examples. Predictions are made on the development and the future of graphical programming a ...

3 A real-time procedural shading system for programmable graphics hardware
Kekoa Proudfoot, William R. Mark, Svetoslav Tzvetkov, Pat Hanrahan
August 2001 **Proceedings of the 28th annual conference on Computer graphics and interactive techniques**
Full text available: pdf(1.20 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Real-time graphics hardware is becoming programmable, but this programmable hardware is complex and difficult to use given current APIs. Higher-level abstractions would both increase programmer productivity and make programs more portable. However, it is challenging to raise the abstraction level while still providing high performance. We have developed a real-time procedural shading language system designed to achieve this goal.

Our system is organized around multiple *computation* ...

Keywords: *graphics hardware, graphics systems, rendering, shading languages*

4 Graphics Programming Using the Core System

R. Daniel Bergeron, Peter R. Bono, James D. Foley

December 1978 **ACM Computing Surveys (CSUR)**, Volume 10 Issue 4

Full text available: pdf(2.92 MB) Additional Information: full citation, references, citings, index terms

5 An object-oriented framework for graphical programming (summary paper)

Steven P. Reiss

June 1986 **ACM SIGPLAN Notices , Proceedings of the 1986 SIGPLAN workshop on Object-oriented programming**, Volume 21 Issue 10

Full text available: pdf(838.85 KB) Additional Information: full citation, references, citings, index terms

6 Large meshes and GPU programming: Brook for GPUs: stream computing on graphics hardware

Ian Buck, Tim Foley, Daniel Horn, Jeremy Sugerman, Kayvon Fatahalian, Mike Houston, Pat Hanrahan

August 2004 **ACM Transactions on Graphics (TOG)**, Volume 23 Issue 3

Full text available: pdf(266.63 KB) mov(21:31 MIN) Additional Information: full citation, abstract, references, citings, index terms

In this paper, we present Brook for GPUs, a system for general-purpose computation on programmable graphics hardware. Brook extends C to include simple data-parallel constructs, enabling the use of the GPU as a streaming co-processor. We present a compiler and runtime system that abstracts and virtualizes many aspects of graphics hardware. In addition, we present an analysis of the effectiveness of the GPU as a compute engine compared to the CPU, to determine when the GPU can outperform the CPU ...

Keywords: Data Parallel Computing, GPU Computing, Brook, Programmable Graphics Hardware, Stream Computing

7 A graphics-based programming-support system

H. P. Frei, D. L. Weller, R. Williams

August 1978 **ACM SIGGRAPH Computer Graphics , Proceedings of the 5th annual conference on Computer graphics and interactive techniques**, Volume 12 Issue 3

Full text available: pdf(1.52 MB) Additional Information: full citation, abstract, references, index terms

A programming support system using extended Nassi-Shneiderman diagrams (NSD) is described. The aim of the work is to develop techniques for improving the quality and cost of specifying, documenting and producing computer programs. NSD's can be executed interpretively or compiled to produce running code. The system implementation has begun and charts can be drawn on a variety of display devices. The system is being developed using the Picture Building System developed earlier.

Keywords: Interactive computer graphics, Nassi-Sheiderman diagrams, Program representation, Structured programming

8 A graphical programming language interface for an intelligent LISP tutor
B. J. Reiser, P. Friedmann, J. Gevins, D. Y. Kimberg, M. Ranney
May 1988 **Proceedings of the SIGCHI conference on Human factors in computing systems**
Full text available: pdf(655.66 KB) Additional Information: full citation, abstract, references, citings, index terms

We describe an intelligent tutor for programming embedded in a graphical programming language. The tutor monitors students' problem solving and provides feedback and guidance. Explanations are generated from the content of the ideal model's problem solving rules. The graphical interface is designed to facilitate the acquisition of causal models of programming. Students work in a medium that corresponds to their planning operations. The interface enables forward and backward chaining, thus c ...

9 Shading and shaders: Efficient partitioning of fragment shaders for multipass rendering on programmable graphics hardware
Eric Chan, Ren Ng, Pradeep Sen, Kekoa Proudfoot, Pat Hanrahan
September 2002 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware**
Full text available: pdf(337.34 KB) Additional Information: full citation, abstract, references, citings, index terms

Real-time programmable graphics hardware has resource constraints that prevent complex shaders from rendering in a single pass. One way to virtualize these resources is to partition shading computations into multiple passes, each of which satisfies the given constraints. Many such partitions exist for a shader, but it is important to find one that renders efficiently. We present Recursive Dominator Split (RDS), a polynomial-time algorithm that uses a cost model to find near-optimal partitions of ...

Keywords: graph partitioning algorithms, multipass rendering, programmable graphics hardware, shading languages

10 Ray tracing on programmable graphics hardware
Timothy J. Purcell, Ian Buck, William R. Mark, Pat Hanrahan
July 2002 **ACM Transactions on Graphics (TOG) , Proceedings of the 29th annual conference on Computer graphics and interactive techniques**, Volume 21 Issue 3
Full text available: pdf(454.93 KB) Additional Information: full citation, abstract, references, citings, index terms

Recently a breakthrough has occurred in graphics hardware: fixed function pipelines have been replaced with programmable vertex and fragment processors. In the near future, the graphics pipeline is likely to evolve into a general programmable stream processor capable of more than simply feed-forward triangle rendering.In this paper, we evaluate these trends in programmability of the graphics pipeline and explain how ray tracing can be mapped to graphics hardware. Using our simulator, we analyze ...

Keywords: programmable graphics hardware, ray tracing

11 Learning, research, and the graphical representation of programming
Robert P. Taylor, Nancy Cunniff, Minh Uchiyama
November 1986 **Proceedings of 1986 ACM Fall joint computer conference**
Full text available: pdf(789.06 KB) Additional Information: full citation, references, index terms

12 Why looking isn't always seeing: readership skills and graphical programming
Marian Petre
June 1995 **Communications of the ACM**, Volume 38 Issue 6

Full text available: pdf(631.72 KB) Additional Information: full citation, abstract, references, citings, index terms

Many believe that visual programming techniques are quite close to developers. This article reports on some fascinating research focusing on understanding how textual and visual representations for software differ in effectiveness. Among other things, it is determined that the differences lie not so much in the textual-visual distinction as in the degree to which specific representations support the conventions experts expect.

13 HDR and tone mapping: Interactive time-dependent tone mapping using programmable graphics hardware
Nolan Goodnight, Rui Wang, Cliff Woolley, Greg Humphreys
June 2003 **Proceedings of the 14th Eurographics workshop on Rendering EGRW '03**

Full text available: pdf(7.56 MB) Additional Information: full citation, abstract, references, citings, index terms

Modern graphics architectures have replaced stages of the graphics pipeline with fully programmable modules. Therefore, it is now possible to perform fairly general computation on each vertex or fragment in a scene. In addition, the nature of the graphics pipeline makes substantial computational power available if the programs have a suitable structure. In this paper, we show that it is possible to cleanly map a state-of-the-art tone mapping algorithm to the pixel processor. This allows an inter ...

14 Session II: Programming graphics processors functionally
Conal Elliott
September 2004 **Proceedings of the ACM SIGPLAN workshop on Haskell**

Full text available: pdf(673.50 KB) Additional Information: full citation, abstract, references, index terms

Graphics cards for personal computers have recently undergone a radical transformation from fixed-function graphics pipelines to multi-processor, programmable architectures. Multi-processor architectures are clearly advantageous for graphics for the simple reason that graphics computations are naturally concurrent, mapping well to stateless stream processing. They therefore parallelize easily and need no random access to memory with its problematic latencies.This paper presents *Vertigo*, a ...

Keywords: 3D modeling, code generation, compilers, computer algebra, computer graphics, domain-specific languages, functional geometry, functional programming, graphics languages, graphics processors, partial evaluation, procedural geometry, procedural shading, shading languages

15 GIDTS: a graphical programming environment for Prolog
Gabriella Kókai, Jörg Nilson, Christian Niss
September 1999 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 1999 ACM SIGPLAN-SIGSOFT workshop on Program analysis for software tools and engineering**, Volume 24 Issue 5

Full text available: pdf(1.47 MB) Additional Information: full citation, abstract, references, index terms

This paper puts forward the Graphical Interactive Diagnosing, Testing and Slicing System (*GIDTS*) which is a graphical programming environment for PROLOG programs. The *IDTS*part of the system integrates Shapiro's Interactive Diagnosis Algorithm with the Category Partition Testing Method (*CPM*) and a slicing technique performing the *algorithmic debugging* and *functional testing* of PROLOG programs. The integration of *IDTS* with a graphical user interface (*GU ...*

16 Graphics Programming Using a Database System with Dependency Declarations
M. T. Garrett, J. D. Foley
April 1982 **ACM Transactions on Graphics (TOG)**, Volume 1 Issue 2
Full text available: pdf(1.26 MB) Additional Information: full citation, references, citings, index terms

Keywords: dependencies, nonprocedural programming, production rules

17 Shading, surfaces, and collision detection: Mesh mutation in programmable graphics hardware
Le-Jeng Shiue, Vineet Goel, Jorg Peters
July 2003 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware**
Full text available: pdf(789.95 KB) Additional Information: full citation, abstract, references, citings

We show how a future graphics processor unit (GPU), enhanced with random read and write to video memory, can represent, refine and adjust complex meshes arising in modeling, simulation and animation. To leverage SIMD parallelism, a general model based on the mesh atlas is developed and a particular implementation without adjacency pointers is proposed in which primal, binary refinement of, possibly mixed, quadrilateral and triangular meshes of arbitrary topological genus, as well as their traver ...

18 Declarative programming of graphical interfaces by visual examples
Ken Miyashita, Satoshi Matsuoka, Shin Takahashi, Akinori Yonezawa, Tomihisa Kamada
December 1992 **Proceedings of the 5th annual ACM symposium on User interface software and technology**
Full text available: pdf(1.26 MB) Additional Information: full citation, abstract, references, citings, index terms

Graphical user interfaces (GUI) provide intuitive and easy means for users to communicate with computers. However, construction of GUI software requires complex programming that is far from being intuitive. Because of the "semantic gap" between the textual application program and its graphical interface, the programmer himself must conceptually maintain the correspondence between the textual programming and the graphical image of the resulting interface. Instead, we propose a pr ...

Keywords: constraints, direct manipulation, graphical user interface, layouts, programming by example, visualization

19 Simulation and computation: A multigrid solver for boundary value problems using programmable graphics hardware
Nolan Goodnight, Cliff Woolley, Gregory Lewin, David Luebke, Greg Humphreys
July 2003 **Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware**
Full text available: pdf(2.80 MB) Additional Information: full citation, abstract, references, citings, index terms

We present a case study in the application of graphics hardware to general-purpose numeric computing. Specifically, we describe a system, built on programmable graphics hardware, able to solve a variety of partial differential equations with complex boundary conditions. Many areas of graphics, simulation, and computational science require efficient techniques for solving such equations. Our system implements the **multigrid method**, a fast and popular approach to solving large boundary value ...

20 A graphical programming environment in Ada
Jorge L. Diaz-Herrera, Shawna C. Gregory
March 1986 **Proceedings of the third annual Washington Ada symposium on Ada: Ada use in focus : practical lessons in perspective**
Full text available: pdf(632.20 KB) Additional Information: full citation, references, citings

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next